Аль Шарари Мохаммад Ибрагим – аспирант, Кафедра Электроснабжения, Институт энергосбережения и энергоменеджмента, Национальный технический университет Украины, «Киевский политехнический институт», ул. Борщаговская, 115, г. Киев, Украина, 03056, e-mail: mr_sharari@yahoo.com

Alsharari Mohammad – postgraduate student, Department of Electricity supply, Institute of energy saving and energy management, National Technical University of Ukraine "Kyiv Polytechnic Institute", 115 Borschagivska Str., Kiyv, Ukraine, 03056, tel.: (093) 289-33-42, e-mail: mr_sharari@yahoo.com

Чекамова Вікторія Вікторівна – магістр, кафедра Електропостачання, Інститут енергозбереження та енергоменеджменту, Національний технічний університет України «Київський політехнічний інститут», вул. Борщагівська, 115, м. Київ, Україна, 03056, e-mail: vika.chekamova@gmail.com

Чекамова Виктория Викторовна – магистр, Кафедра Электроснабжения, Институт энергосбережения и энергоменеджмента, Национальный технический университет Украины, «Киевский политехнический институт», ул. Борщаговская, 115, г. Киев, Украина, 03056, , e-mail: vika.chekamova@gmail.com

Chekamova Viktoriia – master, Department of Electricity supply, Institute of energy saving and energy management, National Technical University of Ukraine "Kyiv Polytechnic Institute", 115 Borschagivska Str., Kiyv, Ukraine, 03056, e-mail: vika.chekamova@gmail.com

УДК 004.912

О. Б. КУНГУРЦЕВ, О. А. БЛАЖКО, С. В. КОВАЛЬЧУК, М.О. СКРИПКІН

АВТОМАТИЗАЦІЯ СТВОРЕННЯ СХОВИЩА ДАНИХ ЕЛЕКТРОННИХ ДОКУМЕНТІВ З ВЕБ-ПОРТАЛІВ ВІДКРИТИХ ДАНИХ

Розглядається процес створення сховища даних електронних документів національного Веб-порталу відкритих даних України. Для скорочення трудомісткості процесу запропоновано розвиток методу порівняння текстів шляхом визначення інтегральної близькості структурованих текстів та їх елементів у вигляді рядків і стовпців, що дозволяє автоматизувати процес встановлення зв'язку між наборами даних при створенні сховища. Програмне забезпечення методу апробовано на наборах відкритих даних національного Веб-порталу. Результати дослідження можуть бути використані при створенні сховищ даних в системах електронного документообігу.

Ключові слова: електронні документи, відкриті данні, зв`язані данні, синтаксичний аналізатор, сховище даних.

Рассматривается процесс создания хранилища данных электронных документов национального Веб-портала открытых данных Украины. Для сокращения трудоемкости процесса предложено развитие метода сравнения текстов путем определения интегральной близости структурированных текстов и их элементов в виде строк и столбцов, что позволяет автоматизировать процесс установления связи между наборами данных при создании хранилища. Программное обеспечение метода апробировано на наборах открытых данных национального Веб-портала. Результаты исследования могут быть использованы при создании хранилищ данных в системах электронного документооборота.

Ключевые слова: электронные документы, открытые данные, связанные данные, синтаксический анализатор, хранилище данных.

The paper considers the process of creating of electronic documents in data warehouse on the web-portal of open data.The result of this study is the method of text comparison for two structured electronic documents, which presented in tabular form to determine the possibility of their association in the data warehouse. The Scientific novelty of work is improvement of method for comparing the texts with integrated proximity of structured texts and their elements in rows and columns in a table, which allows to automate the process of establishing a semantic link between the data sets to create a data warehouse. The software of proposed method is approved by the example of the DOC-format documents on the web-site of the Main Statistical Office in the Odessa region, which stores the operational statistics of socio-economic development of the region. For the experiments was created a public Web-portal of open data at the Odessa area on the basis of free software DKAN. As a result of automated analysis of documents with data tables were created open data sets. For more than half sets automatically semantic links have been established and carried out the union of these sets into a single data warehouse. It will allow a more qualitative analytical assessment of socio-economic processes using diagrams and cartographic type of visualization. The results of work can be used to create any kind of data warehouse in electronic document management systems.

Keywords: electronic documents, open data, linked data, data parser, data warehouse.

Вступ. На початку 2015 року з появою Закону «Про внесення змін до деяких законів України щодо доступу до публічної інформації у формі відкритих даних» держава приєдналася до всесвітнього процесу структуризації публічної інформації. Закон передбачає централізоване розміщення публічної інформації на Веб-порталах у формі електронних даних. За рік до появи цього Закону силами громадських організацій із грантовою підтримкою з`явився національний портал за адресою *http://data.gov.ua*, на якому було створено 100 наборів даних у 12 категоріях. Нажаль, поява Закону не стала поштовхом до відкриття даних і на порталі з`явилося лише декілька наборів. Наступна Постанова Кабінету Міністрів України № 835 від 21.10.2015 «Про затвердження Положення про набори даних, які підлягають оприлюдненню у формі відкритих даних» [1] зобов`язала державні установи розмістити публічні дані на порталі, надавши перевагу структурованим текстовим форматам *CSV/XML/JSON* перед слабоструктурованими форматами *DOC(X)/XLS(X)/PDF*. Після виходу Постанови процес не прискорився і менше половини наборів так і залишається у нерекомендованих неструктурованих форматах, що не дозволяє надати *API*-доступ у форматі *XML*/*JSON* для Веб/мобільних застосувань у соціально-економічних сферах, збільшуючи вартість їх супроводу та, відповідно, підвищуючи ризик зриву планів Кабінету Міністрів із реалізації *IT*-проектів створення *E*-сервісів на основі відкритих даних. Основною причиною низької якості у відкритті даних є:

1) значна трудомісткість ручного процесу перетворення даних з документів офісних систем у *CSV*-формат та наявність помилок користувачів, які пов'язані з різними форматами зберігання, кодування та структурами таблиць;

2) відсутніть автоматизованого процесу створення сховища даних наборів відкритих даних, яке зв`язує набори даних за семантичними зв`язками між наборами даних з метою отримання нової інформації та інтелектуальної аналітичної обробки.

Якщо першу проблему частково можна вирішити організаційними засобами через створення методичного забезпечення та проведення тренінгів з державними службовцями, то другу проблему можна вирішити лише технічно, що і стало темою цієї роботи.

Аналіз літературних даних та постановка проблеми. Відомо, що сховища даних створюються з урахуванням таких особливостей [2]: джерел даних у формі традиційних систем реєстрації операцій, електронних документів або наборів даних, а також операцій з даними на рівні вилучення, перетворення, завантаження, аналізу і представлень результатів аналізу. У розподіленому середовищі зберігання джерел даних, яким є Веб-середовище, однією з основних проблем є висока трудомісткість створення словників метаданих, що дозволяють комп'ютерам і користувачам розуміти структури даних, на основі яких будуть прийматися рішення в процесі використання сховищ даних.

Але говорячи про автоматизацію створення сховищ даних, на першому місці стоїть підтримка метаданих у формі, придатній для комп'ютерної обробки, що і стало причиною створення *RDF*-моделі представлення даних (англ. *Resource Description Framework*) [3]. Модель представлена висловлюваннями у вигляді триплетів «суб'єкт-предикат-об'єкт», які об'єднуються в семантичний граф. Але *RDF* - лише модель, тому для її ефективної програмної реалізації була запропонована архітектура порталів відкритих даних [4].

На рисунку 1 представлено ресурси та процеси, які повинні супроводжувати розвиток порталів відкритих даних в Україні для інформаційної підтримки державних *E*-сервісів та комерційних Веб/мобільних застосувань у соціально-економічних сферах.

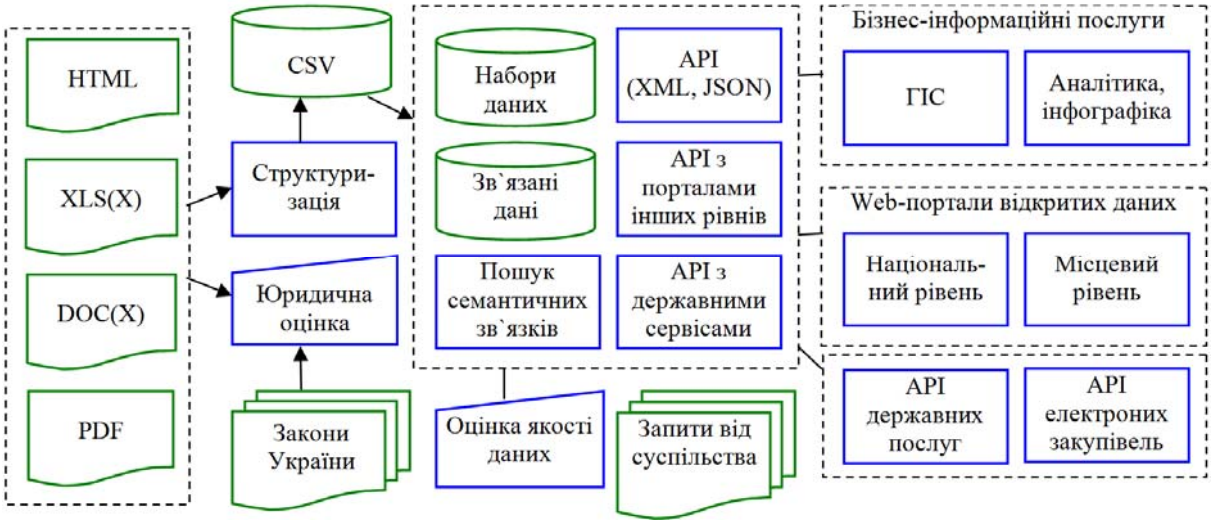

Рис. 1 – Ресурси та процеси із супроводу порталів відкритих даних

Програмне забезпечення Веб-порталів відкритих даних більшості країн світу створено на основі вільних продуктів: *CKAN* (*http://ckan.org*), наприклад, США (*http://data.gov*), Великобританія (*http://data.gov.uk*); *Socrata* (http://socrata.com), наприклад, НьюЙорк (*https://data.ny.gov*), Вашингтон (*https://data.wa.gov*), Сан Франциско (*https://data.sfgov.org*); *DKAN* (*http://nucivic.com/dkan*), наприклад, Італія (*http://dati.gov.it*). Зазначені продукти при реєстрації наборів відкритих даних вимагають створення їх паспортів, колонки яких відповідають структурі словника каталогу даних *DCAT* (англ. *Data Catalog Vocabulary*) [5], спроектованого з урахуванням *RDF*-моделі. Структура *DCAT* включає: ідентифікаційний номер набору даних; найменування набору даних; стислий опис змісту набору даних; відомості про мову інформації, яка міститься у наборі даних; формат (формати), в якому доступний набір даних; формат стиснення набору даних (за наявності такого стиснення); дата і час першого оприлюднення набору даних; дата і час внесення останніх змін до набору даних; дата актуальності даних у наборі даних; періодичність оновлення набору даних; ключові слова, які відображають основний зміст набору даних; гіперпосилання на набір даних (електронний документ для завантаження або *API*-інтерфейс); гіперпосилання на структуру набору даних (електронний документ для завантаження або *API*-інтерфейс); відомості про розпорядника інформації, у володінні якого перебуває набір даних; відомості про відповідальну особу з питань доступу до публічної інформації розпорядника інформації, яка відповідає за оприлюднення інформації та адреса її електронної пошти.

Створене програмне забезпечення щодо заповнення метаданих сховища наборів відкритих даних в *DCAT*-форматі дозволяє вручну заповнювати сховище без автоматизації процесу встановлення семантичних зв'язків між наборами даних. Властивість зв'язності даних (англ. *Linked Data*) у вигляді колекції взаємопов'язаних наборів даних в Інтернеті або опису методів публікації пов'язаних між собою структурованих даних було розглянуто в роботі [6] з узагальненням на технологічному рівні в роботі [7]. В роботі [8] процес встановлення зв'язності віднесений до задачі створення інструментів обробки природної мови, що стикається зі складнощами усунення неоднозначності сенсу слів для узгодження ресурсів різних наборів даних.

В роботі [9] розглядається метод визначення бли-

зькості контентів, пов'язаних з Веб-форумами. Однак автори вказують на наявність попередньо виявлених ключових слів. В той же час, аналіз паспортів наборів даних на Веб-порталах показує, що колонка з ключовими словами заповнена частково і тільки у вигляді одного слова, що не дозволяє використати вказаний метод. В роботі [10] розглянуто окремий випадок визначення близькості подій в новинному потоці для їх подальшого злиття, але у випадку з наборами даних немає необхідності виділяти події, тому для випадку с веб-порталом відкритих даних критерій близькості повинен визначатися іншим способом. Таким чином, створення нових методів встановлення зв'язку між наборами даних із ресурсами *CSV*-формату при створенні сховища даних на Веб-порталах відкритих даних є актуальною задачею.

Ціль та задачі дослідження. Метою роботи є автоматизація процесу встановлення зв'язку між наборами даних із ресурсами *CSV*-формату при створенні сховища даних з метою підвищення ефективності їх подальшої аналітичної обробки.

Для досягнення мети пропонується вирішити наступні задачі:

- виділити в наборах даних терміни з аналізованих текстів ресурсів *CSV*-формату (побудувати словник термінів);
- визначити близькість *CSV*-таблиць по кількості однакових термінів;
- виявити в *CSV*-таблицях з певним ступенем близькості подібні рядки і стовпці.

Виділення термінів з аналізованих текстів наборів даних. Найбільш інформативними словами в реченні є іменники та іменні групи. Саме вони складають основу термінологічної бази предметної області і повинні бути включені в словник. Термін – це слово, стійке словосполучення або скорочення, яке виражає і певною мірою класифікує в даній предметній області певне поняття чи сутність, відображаючи у своїй смисловій структурі характерні ознаки об'єкту і взаємозв'язку цього об'єкта з іншими з достатньою для взаємного спілкування точністю [11].

Відомі рішення побудови словників орієнтовані на англо-німецьку групу мов [12]. В роботах [13-15], орієнтованих на групи слов'янських мов, відсутня автоматизована класифікація текстів, визначення синонімів і значень термінів. В роботі [16] задача обмежена рамками спілкування користувача на природній мові з реляційною базою даних.

Для виділення термінів запропоновано проведення синтаксичного аналізу текстів документів з використанням програмного забезпечення з відкритим вихідним кодом *Language Tool* з модулем *language-uk* [17] та *Cognitive Dwarf* [18].

Для побудови словника термінів передбачено два проходи по тексту.

У процесі першого проходу виконуються наступні дії:

1) розбивка тексту на слова;
2) видалення слів із списку, які повинні ігноруватись;
3) вибір окремого слова, коли проводиться морфологічний розбір слова. і, якщо слово є іменником, виконуємо для нього пункт 5, а якщо слово не іменник, то перехід до пункту 3;
4) перевірка чи є вхідне слово унікальним для результуючого списку, і якщо так, то добавляємо його у список, а якщо ні, то збільшуємо значення змінної, яка накопичує кількість входжень слова у текст, на 1;
5) якщо у вхідному спискові є ще слова, то перехід до пункту 3;
6) іменники, які рідко зустрічалися у тексті, виключаються зі словника.

Під час другого проходу для кожного іменника із словника будується комбінація слів, які оточують його зліва та справа. Підраховується кількість входження в текст для кожної комбінації слів.

Словосполучення з високою частотою входження (наприклад, рівним 50% від частоти іменника) вживаються як терміни.

Встановлення зв'язності документів на основі синтаксичного аналізу текстів та знайдених термінів. Таким чином, після аналізу вхідного тексту T та побудови словника термінів, отримуємо текст з результатом $Tres$, який містить в собі всі терміни в початковій формі з вказаною кількістю повторень в тексті для кожного.

1) Близькість документів на основі порівнянь їхніх термінів

Представимо текст з результатом $Tres$ множиною записів:

$$Tres = \{term_i\} i = 1, n \,, \tag{1}$$

де $term_i$ – запис з унікальним терміном; n – кількість унікальних термінів в тексті $Tres$.Кожен запис $term_i$ представляє собою кортеж:

$$term_i = < tx_i, m_i > , \tag{2}$$

де tx_i – термін (одне або декілька слів); m_i – кількість повторень терміну.

Масив у з усіма результатами $Tres$ має вигляд:

$$AllTres = \{Tres_j\} j = 1, k \,. \tag{3}$$

Для кожного результату $Tres$ можна підрахувати загальну кількість термінів N :

$$N_j = \sum_{i=0}^{n} m_i \,. \tag{4}$$

Кількість порівнянь між текстами $Tres_j$ в масиві $AllTres$ становить $k*(k-1)$.

Здійснюємо перевірку кожен з кожним терміном між різними документами $Tres$, і знаходимо спільні терміни враховуючи меншу спільну кількість m_i .

Кількість порівнянь між двома документами становить добутку кількості унікальних термінів в текстах $Tres$, ($n \in Tres_j * n \in Tres_{j+1}$).

Якщо термін з першого тексту співпадає з терміном іншого, тобто виконується умова $tx_i \in Tres_j = tx_i \in Tres_{j+1}$, порівнюємо кількість повто-

рень даного терміну в двох документах і обираємо меншу (якщо $m_i \in Tres_j \le m_i \in Tres_{j+1}$, то результат $m_i \in Tres_j$, якщо $m_i \in Tres_j > m_i \in Tres_{j+1}$, то результат $m_i \in Tres_{j+1}$). Після цього здійснюємо порівняння першого з іншими і т.д.

Всі порівняння заносимо в масив, результат порівняння має наступний вигляд:

$$Buf = \{res_z\} z = 1, x . \quad (5)$$

Кожен запис представляє собою кортеж:

$$res_z =< tx_z, m_z > . \quad (6)$$

Після закінчення перевірки між двома документами, здійснюємо обрахунок усіх спільних термінів між ними *Nres* (*Nres* є спільним для двох документів *Tres* , які порівнювалися між собою):

$$Nres = \sum_{z=0}^{x} m_z . \quad (7)$$

У першому випадку за допомогою формул (4), (7) знаходимо відсоток повторень в кожному з двох аналізованих текстів без урахування ваги термінів $proc_1$:

$$proc_1 = \frac{Nres}{N} * 100\% . \quad (8)$$

Отже, для першого документа відносно другого відсоток буде становити:

$$proc_1 \in Tres_j = \frac{Nres}{N_j} * 100\% .$$

Для другого документа відносно першого відсоток буде становити:

$$proc_1 \in Tres_{j+1} = \frac{Nres}{N_{j+1}} * 100\% .$$

Формуємо результат у вигляді:

$Tres_j(Tres_{j+1}) = proc \in Tres_j$,

$Tres_{j+1}(Tres_j) = proc \in Tres_{j+1}$.

Недоліком є не врахування кількості повторень окремих термінів, так як враховується тільки загальна кількість повторень.

Варіант з урахування ваги терміну. Чим більша кількість повторень терміну тим вищий його відсоток відносно інших термінів. Потрібно здійснити обрахунок для кожного терміну і просумувати отримані відсотки.

Відсоток повторення для кожного терміну формується на підставі аналізу результатів формули (7):

$$procTerm_z = \frac{m_z}{N} * \frac{m_z}{Nres} * 100\% , \quad (9)$$

де $\frac{m_z}{N}$ – відношення кількості повторень унікального терміну відносно усіх знайдених термінів в $Tres_j$;

$\frac{m_z}{Nres}$ – відношення кількості повторень унікального терміну відносно усіх спільних термінів між $Tres_j$ та $Tres_{j+1}$.

Наступним етапом сумуємо всі відсотки унікальних термінів, де $proc_2$ і буде відсотком близькості документів з урахуванням ваги термінів:

$$proc_2 = (procTerm_z + procTerm_{z+1} ... procTerm_x) * kof , \quad (10)$$

де *kof* – коефіцієнт кількості спільних унікальних термінів.

Коефіцієнт *kof* обраховується з урахуванням кількості унікальних термінів в масиві *Buf* (5):

$$kof = \sqrt{x} , \quad (11)$$

де x – кількість унікальних термінів в *Buf* .

2) Виявлення подібних стовпців і рядків в таблицях

Після виявлення близькості текстів по термінах, потрібно здійснити перевірку по стовпцях і рядках двох текстів T_j і T_{j+1}. Для цього виявляємо всі таблиці в цих документах і по кожній з них записуємо усі заголовки рядків і стовпців.

Уявімо собі результат аналізу тексту T у вигляді множини знайдених таблиць *Ttable* :

$$Ttable = \{table_y\} y = 1, u , \quad (12)$$

де $table_y$ – знайдена таблиця в тексті T .

Кожна знайдена таблиця представляє собою кортеж:

$$table_y =< rows_y, columns_y > , \quad (13)$$

де $rows_y$ – масив записів усіх заголовних рядків в таблиці $table_y$; $columns_y$ – масив записів усіх заголовних стовпців в таблиці $table_y$.

Наступним кроком здійснюється пошук термінів в кожному заголовку рядків і стовпців відповідно до попередніх результатів в *Tres* . Після цього формуються масиви $rows_y$ і $columns_y$.

Представимо, що $rows_y$ складається з множини заголовків рядків:

$$rows_y = \{row_q\} q = 1, w , \quad (14)$$

де row_q – масив термінів *tx* для заголовка рядка.

Відповідно до цього для $columns_y$ отримуємо:

$$columns_y = \{column_e\} e = 1, r , \quad (15)$$

де $column_e$ – масив термінів *tx* для заголовка стовпця.

Отже, row_q має вигляд:

$$row_q = \{tx_a\} a = 1, b \,. \quad (16)$$

Відповідно $column_e$ має наступний вигляд:

$$column_e = \{tx_f\} f = 1, g \,. \quad (17)$$

Усі таблиці $table_y$, двох порівнювальних текстів T, порівнюються кожна з кожною по стовпцях і рядках. Для кожного заголовку рядка рахується відсоток близькості з іншим рядком, відповідні дії відбуваються і з заголовками стовпців. Якщо в заголовку рядка або стовпця немає термінів ($b = 0, g = 0$), то обрахунок по формулі не здійснюється.

Відсоток близькості рахується наступним чином:

$$p = \frac{com}{mid} * 100\%, \quad (18)$$

де com – кількість спільних термінів між $row_q \in Ttable_j$ та $row_q \in Ttable_{j+1}$ для рядка, між $column_e \in Ttable_j$ та $column_e \in Ttable_{j+1}$ для стовпця;

mid – середня кількість термінів між $row_q \in Ttable_j$ та $row_q \in Ttable_{j+1}$ для рядка, між $column_e \in Ttable_j$ та $column_e \in Ttable_{j+1}$ для стовпця;.

Знайшовши відсоток p для кожного заголовку рядка та стовпця, здійснюємо обрахунок середнього відсотка близькості $pMid$ заголовного рядка або стовпця:

$$pMid = (\sum p) / \arg, \quad (19)$$

де arg – кількість не нульових порівнянь між заголовками рядків або стовпців.

Знайшовши середнє значення по заголовках рядків і стовпців, взнаємо значення близькості порівнювальних таблиць $table_y \in Ttable_j$ та $table_y \in Ttable_{j+1}$:

$$TableP = \frac{pMid_{row} + pMid_{column}}{2}. \quad (20)$$

Якщо ні одного збігу не було виявлено, то зв'язок між порівнювальними таблицями становить 0.

В результаті знайшовши всі близькості між таблицями двох порівнювальних документів можемо обрахувати загальну близькість між документами T_j та T_{j+1}:

$$proc_3 = (\sum_{p=1}^{k} TableP_p) / k, \quad (21)$$

де k – кількість порівнянь між таблицями двох документів, становить добутку кількості таблиць в двох документах відповідно до формули (12) $k = u \in Ttable_j * u \in Ttable_{j+1}$.

3) Додання нових термінів до словника синтаксичного аналізатора. Аналіз документів показав, що словник *language tool* не містить ряд слів, які зустрічаються у документах. Найчастіше це особисті назви. Для вирішення вказаної проблеми розглянуто додання слів у файл із розширенням *.txt* в кодуванні *UTF*-8. Формат додання має наступний порядок: слово; початкова форма слова; інформація про слово. Недоліком такого рішення є те, що пошук по такому файлові буде здійснюватися занадто довго, і його розмір також буде великим при великій кількості слів.

Іншим рішенням є створення власного словника. Перевагою є набагато менший розмір і відповідно швидкість перегляду. Словник створюється за допомогою засобів *language tool*, шляхом застосування інструмента із вхідними параметрами: текстовим файлом (як у першому способі), і файлом у якому буде службова інформація (файл, який є у *language tool*). Необхідно тільки 2 інструменти і відповідно 2 їх виклики. За допомогою даного підходу реалізовано розширення словника синтаксичного аналізатора за рахунок назв із певних предметних областей, наприклад: назв районів, міст, областей.

Висновки. Запропонований в роботі розвиток методу порівняння текстів шляхом визначення інтегральної близькості структурованих текстів та їх елементів у вигляді рядків і стовпців *CSV*-таблиці із ресурсів різних наборів відкритих даних форматів *DOC(X)* було апробовано після включення його програмного забезпечення у вигляді 5-го модуля до програмного комплексу, який містить модулі: (1) представлення електронного документу в *HTML*-форматі, (2) структурно-семантичний аналіз документа, (3) створення *CSV*-таблиці, (4) створення паспорту набору, (5) пошук зв`язків між поточним набором та наборами зі сховища з подальшим оновленням сховища. Наприкінці листопада 2015 року було оновлено програмне забезпечення національного Веб-порталу та очищено сховище наборів даних. Тому апробацію було проведено на прикладі документів з сайту Головного управління статистики в Одеській області, розміщеного за адресою http://od.ukrstat.gov.ua/, у розділі «Експрес-випуски» за 2015 рік. Проаналізовано 65 документів *DOC*-формату, які містили 143 таблиці з даними за такими категоріями: освіта, ринок праці, зайнятість та безробіття, оплата праці та соціально-трудові відносини, соціальний захист, економічна діяльність, будівництво, внутрішня торгівля, діяльність підприємств, енергетика, капітальні інвестиції, навколишнє середовище, послуги, наука, промисловість, сільське, лісове та рибне господарство, транспорт, туризм, зовнішньоекономічна діяльність, національні рахунки, ціни. В результаті було створено 143 набори відкритих даних, з яких для 87 було автоматично встановлено семантичні зв`язки на рівні відкритих даних районів області. Знайдені набори створили сховище даних, яке дозволить проводити більш якісну аналітичну оцінку соціально-економічних процесів області з використанням діаграм та картографічних засобів візуалізації.

Автори даної роботи входять у волонтерську групу зі створення громадського Веб-порталу відкри-

тих даних Одеської області за адресою *http://data.ngorg.od.ua* на основі вільного програмного продукту *DKAN*, особливістю якого є фінансово-економічна оренда *PHP-hosting*-серверів. Тому всі створені набори було розміщено на вказаному порталі.

Список літератури:

1. Про затвердження Положення про набори даних, які підлягають оприлюдненню у формі відкритих даних [Електронний ресурс]: Постанова Кабінету Міністрів України від 21.10.2015 № 835. – Режим доступу: \www/URL: http://zakon3.rada.gov.ua/laws/show/835-2015-%D0%BF
2. *Барсегян, А. А.* Методы и модели анализа данных: OLAP и Data Mining [Текст] / *А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод*. – Санкт-Петербург: БХВ-Петербург, 2004. – 336 с.
3. RDF 1.1 Primer [Electronic resource] / W3C Working Group Note. – Available at: \www/URL: https://www.w3.org/TR/rdf11-primer/. – 24.06.2014.
4. *Auer, S.* DBpedia: A nucleus for a web of open data [Text]: International Semantic Web Conference / *S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. Ives* // Lecture Notes in Computer Science. − 2007. − P. 722–735. doi:10.1007/978-3-540-76298-0_52
5. Data Catalog Vocabulary (DCAT) [Electronic resource] / W3C Working Group Note. – Available at: \www/URL: https://www.w3.org/TR/vocab-dcat/. – 16.01.2014.
6. *Bizer, C.* Linked Data − the story so far [Text] / *C. Bizer, T. Heath, T. Berners-Lee* // International Journal on Semantic Web and Information Systems. − 2009. – Vol. 5, № 3. – P. 1–22. doi:10.4018/jswis.2009081901
7. *Wood, D.* Linked Data. Structured Data on the Web [Text] / *D. Wood, M. Zaidman, L. Ruth. M. Hausenblas.* – Manning, 2014. − 276 p.
8. *McCrae, J. P.* Reconciling Heterogeneous Descriptions of Language Resources [Text] / *J. P. McCrae, P. Cimiano, V. Rodriguez-Doncel, D. Vila Suero, J. Gracia, L. Matteis, P. Buitelaar* // Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications. − 2015. − P. 39–48. doi:10.18653/v1/w15-4205
9. *Пелещишин, А. М.* Структурування інформаційного наповнення для
1. покращення рангу веб-форуму [Текст] / *А. М. Пелещишин, Ю. О. Сєров, К. О. Слобода* // Східно-Європейський журнал передових технологій. – 2010. – № 6/8 (48). – С. 37–39. − Режим доступу: \www/URL: http://journals.uran.ua/eejet/article/view/5763/5196
10. *Черенков, И. А.* Подход выделения событий в новостном потоке [Текст] / *И. А. Черенков, С. В. Орехов* // Восточно-Европейский журнал передовых технологий. – 2013. – № 1/4 (61). – С. 62−64. − Режим доступа: \www/URL: http://journals.uran.ua/eejet/article/view/9178/7968
11. *Ожегов, С. И.* Толковый словарь русского языка [Текст] / *С. И. Ожегов, Н. Ю. Шведова.* – Москва: Мир, 2011. – 736 с.
12. JaLingo is a free OS independent dictionary application [Electronic resource] / JaLingo. – Available at: \www/URL: http://jalingo.sourceforge.net/. − 11.12.2006.
13. *Кунгурцев, А. Б.* Формирование словаря предметной области [Текст] / *А. Б. Кунгурцев, И. В. Барыкина* // Искусственный интеллект. – 2006. – № 1. – С. 144–151.
14. *Кунгурцев, А. Б.* Применение сетей фреймов для построения модели извлечения фактов из текстов на естественном языке [Текст] / *А. Б. Кунгурцев, С. М. Бородавкин* // Искусственный интеллект. – 2009. – № 4. – С. 202–207.
15. *Кунгурцев, А. Б.* Метод построения словарей предметных областей для извлечения фактов из текстов на естественном языке [Текст] / *А. Б. Кунгурцев, С. Н. Бородавкин, А. П. Голуб* // Восточно-Европейский журнал передовых технологий. – 2010. – № 1/4 (43). – С. 32–36. − Режим доступа: \www/URL: http://journals.uran.ua/eejet/article/view/2550/2355
16. *Кунгурцев, А. Б.* Интерфейс для общения пользователей с информационными системами на естественном языке [Текст] / *А. Б. Кунгурцев, Я. В. Поточняк* // Электротехнические и компьютерные системы. – 2014. – № 14. – С. 74–81.
17. Development API [Electronic resource] / LanguageTool. – Available at: \www/URL: https://www.languagetool.org/development/
18. Программный пакет синтаксического разбора и машинного перевода [Электронный ресурс]. – Режим доступа: \www/URL: http://cs.isa.ru:10000/dwarf/. – 24.04.2011.

Bibliography (transliterated):

1. Pro zatverdzhennia Polozhennia pro nabory danykh, yaki pidliahaiut opryliudnenniu u formi vidkrytykh danykh: Postanova Kabinetu Ministriv Ukrainy vid 21.10.2015 No. 835. Available at: http://zakon3.rada.gov.ua/laws/show/835-2015-%D0%BF
2. Barsehian, A., Kupryianov, M., Stepanenko, V., Kholod, Y. (2004). Metodi y modely analyza dannikh: OLAP y Data Mining. Saint Petersburg: BHV-Peterburh, 336.
3. RDF 1.1 Primer (24.06.2014). W3C Working Group Note. Available at: https://www.w3.org/TR/rdf11-primer/
4. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z. (2007). DBpedia: A Nucleus for a Web of Open Data. Lecture Notes in Computer Science, 722–735. doi:10.1007/978-3-540-76298-0_52
5. Data Catalog Vocabulary (DCAT). (16.01.2014). W3C Working Group Note. Available at: https://www.w3.org/TR/vocab-dcat/. Last accessed: 02.01.2016.
6. Bizer, C., Heath, T., Berners-Lee, T. (2009). Linked Data − The Story So Far. International Journal on Semantic Web and Information Systems, 5 (3), 1–22. doi:10.4018/jswis.2009081901
7. Wood, D., Zaidman, M., Ruth, L., Hausenblas, M. (2014). Linked Data. Structured Data on the Web. Manning, 276.
8. McCrae, J. P., Cimiano, P., Rodriguez-Doncel, V., Vila Suero, D., Gracia, J., Matteis, L., Buitelaar, P. (2015). Reconciling Heterogeneous Descriptions of Language Resources. Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications, 39–48. doi:10.18653/v1/w15-4205
9. Peleshchishin, A., Serov, Iu., Sloboda, K. (2010). Structuring content for improving the rank web forum. Eastern-European Journal of Enterprise Technologies, 6(8(48)), 37−39. Available at: http://journals.uran.ua/eejet/article/view/5763/5196
10. Cherenkov, Y., Orekhov, S. (2013). Approach for extracting events from news stream. Eastern-European Journal Of Enterprise Technologies, 1(4(61)), 62−64. Available at: http://journals.uran.ua/eejet/article/view/9178/7968
11. Ozhegov, S. I., Shvedova, N. Iu. (2011). Tolkovyi slovar' russkogo iazyka. Moscow: Mir, 736.
12. JaLingo is a free OS independent dictionary application. (11.12.2006). JaLingo. Available at: http://jalingo.sourceforge.net/. Last accessed: 02.01.2016.
13. Kunhurtsev, A., Barikyna, Y. (2006). Formyrovanye slovaria predmetnoi oblasty. Iskusstvennyi intellekt, 1, 144–151.
14. Kunhurtsev, A., Borodavkin, S. (2009). Primenenie setei freimov dlia postroeniia modeli izvlecheniia faktov iz tekstov na estestvennom iazyke. Iskusstvennyi intellekt, 4, 202–207.
15. Kunhurtsev, A., Borodavkyn, S., Holub, A. (2010). Method of creation of domains dictionaries for extraction of the facts from texts in the natural language. Eastern-European Journal Of Enterprise Technologies, 1(4(43)), 32−36. Available at: http://journals.uran.ua/eejet/article/view/2550/2355
16. Kunhurtsev, A., Potochniak, Y. (2014). Interfeis dlia obshcheniia pol'zovatelei s informatsionnymi sistemami na estestvennom iazyke. Elektrotehnicheskie i komp'iuternye sistemy, 14, 74–81.
17. Development API. LanguageTool. Available at: https://www.languagetool.org/development/. Last accessed: 02.01.2016.
18. Prohrammny paket syntaksycheskoho razbora y mashynnoho perevoda. Available at: http://cs.isa.ru:10000/dwarf/. Last accessed: 24.04.2011

Надійшла (received) 06.01.2016

Бібліографічні описи / Библиографические описания / Bibliographic descriptions

Автоматизация создания хранилища данных электронных документов с Веб-порталов открытых данных/ А. Б. Кунгурцев, А. А. Блажко, С. В. Ковальчук, М. А. Скрипкин// Вісник НТУ «ХПІ». Серія: Механіко-технологічні системи та комплекси. – Харків : НТУ «ХПІ», 2016. – No 4(1176). – С.31–37. – Бібліогр.: 18назв. – ISSN 2079-5459.

Автоматизація створення сховища даних електронних документів з Веб-порталів відкритих даних/ О. Б. Кунгурцев, О. А. Блажко, С. В. Ковальчук, М. О. Скрипкін// Вісник НТУ «ХПІ». Серія: Механіко-технологічні системи та комплекси. – Харків : НТУ «ХПІ», 2016. – No 4(1176). – С.31–37. – Бібліогр.: 180 назв. – ISSN 2079-5459.

Automating of creation of electronic documents warehouse on web-portals of open data/ A. Kungurtsev, O. Blazhko, S. Kovalchuk, M. Skripkin//Bulletin of NTU "KhPI". Series: Mechanical-technological systems and complexes. – Kharkov: NTU "KhPI", 2016. – No 4 (1176) .– P.31–37. – Bibliogr.: 18. – ISSN 2079-5459.

Відомості про авторів / Сведения об авторах / About the Authors

Кунгурцев Олексій Борисович – кандидат технічних наук, професор, Одеський національний політехнічний університет; професор кафедри системного програмного забезпечення, пр. Шевченка, 1, м. Одеса, Україна, 65044; E-mail: abkun@te.net.ua.

Кунгурцев Алексей Борисович – кандидат технических наук, профессор, Одесский национальный политехнический университет; профессор кафедры системного программного обеспечения, пр. Шевченко, 1, г. Одесса, Украина, 65044; E-mail: abkun@te.net.ua.

Kungurtsev Alexei – PhD in Technical Sciences, Professor of Department of System Software, Odessa National Polytechnic University, Shevchenko avenue, 1, Odessa, Ukraine, 65044; E-mail: abkun@te.net.ua.

Блажко Олександр Анатолійович – кандидат технічних наук, доцент, Одеський національний політехнічний університет; доцент кафедри системного програмного забезпечення, пр. Шевченка, 1, м. Одеса, Україна, 65044; E-mail: blazhko@ieee.org.

Блажко Александр Анатолиевич – кандидат технических наук, доцент, Одесский национальный политехнический университет; доцент кафедры системного программного обеспечения, пр. Шевченко, 1, г. Одесса, Украина, 65044; E-mail: blazhko@ieee.org.

Blazhko Oleksandr – PhD in Technical Sciences, associate professor, Associate Professor of Department of System Software, Odessa National Polytechnic University, Str. Shevchenko avenue, 1, Odessa, 65044, Ukraine; E-mail: blazhko@ieee.org.

Ковальчук Сергій Вікторович – аспірант кафедри системного програмного забезпечення, Одеський національний політехнічний університет; пр. Шевченка, 1, м. Одеса, Україна, 65044; E-mail: serhiy_kovalchuk@mail.ua.

Ковальчук Сергей Викторович – аспирант кафедры системного программного обеспечения, Одесский национальный политехнический университет; пр. Шевченко, 1, г. Одесса, Украина, 65044; E-mail: serhiy_kovalchuk@mail.ua.

Kovalchuk Serhiy – PhD student of Department of System Software, Odessa National Polytechnic University; Shevchenko avenue, 1, Odessa, Ukraine, 65044; E-mail: serhiy_kovalchuk@mail.ua.

Скрипкін Михайло Олександрович – магістрант кафедри системного програмного забезпечення, Одеський національний політехнічний університет; проспект Шевченка, 1, м. Одеса, Україна, 65044; E-mail: mishkinstvo@outlook.com.

Скрипкин Михаил Александрович – магистрант кафедры системного программного обеспечения, Одесский национальный политехнический университет; пр. Шевченко, 1, г. Одесса, Украина, 65044; E-mail: mishkinstvo@outlook.com.

Skripkin Mihailo – master student of Department of System Software, Odessa National Polytechnic University; Shevchenko avenue, 1, Odessa, 65044, Ukraine; E-mail: mishkinstvo@outlook.com.